Amendments to the Claims

Listing of Claims:

1-29. (Cancelled)

30. (Currently amended) A computer-implemented method for determining compliance between a source document structured in accordance with a source schema and a target schema, the method comprising steps of:

receiving a source schema description and a target schema description;

receiving the source document, wherein the source document comprises an ordered tree structure with labeled elements;

identifying all corresponding element types in the source and target schemas for grouping the corresponding element types into element type pairs, wherein the element type is an indication of the content of an element, and wherein the element type further comprises an element tag name for XML schema if the source schema is XML schema;

classifying each element type pair, according to their relationship, into one of a group consisting of: matching, disjointed, and intersecting;

confirming compliance of the source document if it is determined that all element type pairs corresponding to root elements of the source document are classified as matching; and

confirming non-compliance of the source document according to the target schema if it is determined that at least one element type pair corresponding to a root element of the source document is classified as disjointed; and further:

if it is determined that at least one element type pair corresponding to the root element is classified as intersecting, performing steps of:

identifying a subtree of the source document beginning with a topmost element of the source document, the subtree corresponding to the at least one element type pair classified as intersecting;

classifying the topmost element type pair into one of a group consisting of:

2

matching, disjointed, and intersecting;

confirming compliance of the source document if it is determined that the topmost element type pair is classified as matching; and

confirming non-compliance of the source document if it is determined that the topmost element type pair is classified as disjointed; and further:

if it is determined that the topmost element type pair is classified as intersecting, performing steps of:

identifying a current child element pair of the topmost element type pair from the subtree, wherein the current child element pair comprises an element type pair from the source schema and an element type pair from the target schema;

classifying the current child element pair into one of a group consisting of: matching, disjointed, and intersecting;

confirming non-compliance of the source document if the current child element pair is classified as disjointed;

processing a next child element pair from the subtree if the current child element pair is classified as matching, wherein the processing step further comprises comparing element pairs; and

identifying the next child element pair from the subtree as the topmost element pair if the next child element pair does not belong to a same parent node as the current child element pair.

31. (Cancelled)   The method of claim 30 further comprising steps of:

if it is determined that at least one element type pair corresponding to the root element is classified as intersecting, performing steps of:

identifying a subtree of the source document beginning with a topmost element of the source document, the subtree corresponding to the at least one element type pair classified as intersecting;

classifying the topmost element type pair into one of a group consisting of: matching,

disjointed, and intersecting;

confirming compliance of the source document if it is determined that the topmost element type pair are classified as matching; and

confirming non-compliance of the source document if it is determined that the topmost element type pair are classified as disjointed.

32. (Cancelled)

33. (Previously presented) The method of claim 30 wherein the matching classification comprises a relationship between the element type pair such that portions of the source document that are in compliance with respect to the element type in the source schema are also valid with respect to the corresponding element type in the target schema.

34. (Previously presented) The method of claim 30 wherein the disjointed classification comprises a relationship between the element type pair such that portions of the source document that are in compliance with respect to the element type in the source schema are not in compliance with respect to the corresponding element type in the target schema.

35. (Previously presented) The method of claim 30 wherein the intersecting classification comprises a relationship between the element type pair such that some portions of the source document that are in compliance with respect to the element type in the source schema are in compliance with respect to the corresponding element type in the target schema and some portions of the source document that are in compliance with respect to the element type in the source schema are not in compliance with respect to the corresponding element type in the target schema.

36. (Currently amended) The method of claim [[32]] 30 wherein the source document is determined to be non compliant with respect to the target schema if said source document

4

contains child element pairs that are identified as disjointed.

37. (Currently amended) The method of claim [[32]] 30 further comprising a step of:

developing an automaton from the child element pair determined to be intersecting to determine if a portion of the source document corresponding to the subtree is in compliance with the target schema.

38. (Previously presented) The method of claim 30 wherein the source schema and the target schema are one of a group consisting of: a regular expression, a document type definition, a finite state automata, an XML schema and a tree automata.

39. (Previously presented) The method of claim 30 wherein the source document is an XML document.

40. (Previously presented) The method of claim 30 wherein the element types are one of a group consisting of: states in a finite state automaton, element type declarations in an XML schema, programming language types and states in a tree automaton.

41. (Previously presented) The method of claim 30 further comprising a step of:

examining the source document for determining if any portions of the source document have been modified subsequent to the source document being determined to be in compliance and limiting the portions of the source document to be validated to those portions of the source document that have been modified.

42. (Previously presented) The method of claim 41 further comprising a step of:

identifying elements of the source document that have been inserted subsequent to the source document being determined to be in compliance and immediately confirming compliance of those inserted elements.

43. (Previously presented) The method of claim 41 further comprising a step of:

identifying elements of the source document that have been deleted subsequent to the source document being determined to be in compliance and ignoring any deleted elements when processing said source document to determine compliance with the target schema.

44. (Previously presented) The method of claim 30 further comprising:

confirming compliance of the source document with respect to one of either the source schema or target schema when the source schema has been modified such that it is the same as the target schema.

45. (Currently amended) The method of claim [[32]] 30 wherein the identifying, classifying, confirming, processing and identifying steps are performed recursively until the current child element pair is classified as disjointed and then the source document is immediately determined to be non-compliant.

46. (Previously presented) The method of claim 45 wherein the processing step is performed sequentially.

47. (Previously presented) The method of claim 45 wherein the processing step is performed in parallel.

48. (Previously presented) The method of claim 45 wherein the processing step is performed as a combination of parallel and sequential processing.

49. (Previously presented) The method of claim 30 wherein determining compliance comprises ascertaining that the source document can be recast into the target schema without causing any loss of data.

50. (Previously presented) The method of claim 30 wherein the step of receiving the schema descriptions further comprises creating the schema descriptions.

51. (Previously presented) The method of claim 30 wherein the step of receiving the schema descriptions further comprises retrieving the schema descriptions from data storage.

52. – 58. (Canceled)

59. (Currently amended) An information processing system for determining compliance between a source document structured in accordance with a source schema and a target schema, the system comprising:

data storage;

a processor configured for executing software code, the software code, when executed, enabling the processor to:

~~receive~~ create a source schema description and a target schema description;

receive the source document, wherein the source document comprises an ordered tree structure with labeled elements, said ordered true structure comprising a subtree, wherein said subtree is a portion of the source document following a hierarchical structure wherein a topmost element conforms to a parent in a parent-child relationship;

identify all corresponding element types in the source and target schemas for grouping the corresponding element types into element type pairs, wherein the element type is an indication of the content of an element, and wherein the element type further comprises an element tag name for XML schema if the source schema is XML schema;

classify each element type pair, according to their relationship, into one of a group consisting of: matching, disjointed, and intersecting;

confirm compliance of the source document if it is determined that all element type pairs corresponding to root elements of the source document are classified as matching; and

confirm non-compliance of the source document according to the target schema if it is determined that at least one element type pair corresponding to a root element of the source document is classified as disjointed; and the processor is further enabled to:

if it is determined that the topmost element type pair are classified as intersecting, perform steps of:

identify a current child element pair of the topmost element type pair from the subtree, wherein the current child element pair comprises an element type pair from the

source schema and an element type pair from the target schema;

classify the current child element pair into one of a group consisting of: matching,

disjointed, and intersecting;

confirm non-compliance of the source document if the current child element pair

is classified as disjointed;

process a next child element pair from the subtree if the current child element pair

is classified as matching; and

identify the next child element pair from the subtree as the topmost element pair if

the next child element pair does not belong to a same parent node as the current child

element pair.

60. (Currently amended) The system of claim 59 wherein the processor is further enabled to:

if it is determined that at least one element type pair corresponding to the root element is

classified as intersecting, performing steps of:

identify a subtree of the source document beginning with a topmost element of the source

document, the subtree corresponding to the at least one element type pair classified as

intersecting;

classify the topmost element type pair into one of a group consisting of: matching,

disjointed, and intersecting;

confirm compliance of the source document if it is determined that the topmost element

type pair are is classified as matching; and

confirm non-compliance of the source document if it is determined that the topmost

element type pair are is classified as disjointed.

61. (Cancelled)

62. (Cancelled)

63. (Cancelled)

64. (Currently amended) The system of claim [[63]] <u>60</u> wherein the processor is further enabled to:

      perform the identify, classify, confirm, process and identify steps recursively until the current child element pair is classified as disjointed and then the source document is immediately determined to be non-compliant.

65. (Currently amended) A computer readable medium comprising software instructions for enabling a processor to:

      receive a source schema description and a target schema description;

      receive the source document, wherein the source document comprises an ordered tree structure with labeled elements;

      identify all corresponding element types in the source and target schemas for grouping the corresponding element types into element type pairs, wherein the element type is an indication of the content of an element, and wherein the element type further comprises an element tag name for XML schema if the source schema is XML schema;

      classify each element type pair, according to their relationship, into one of a group consisting of: matching, disjointed, and intersecting;

      confirm compliance of the source document if it is determined that all element type pairs corresponding to root elements of the source document are classified as matching; and

      confirm non-compliance of the source document according to the target schema if it is determined that at least one element type pair corresponding to a root element of the source document is classified as disjointed <u>and further:</u>

      <u>if it is determined that at least one element type pair corresponding to the root element is classified as intersecting, perform steps of:</u>

            <u>identify a subtree of the source document beginning with a topmost element of the</u>

source document, the subtree corresponding to the at least one element type pair classified as intersecting;

classify the topmost element type pair into one of a group consisting of: matching, disjointed, and intersecting;

confirm compliance of the source document if it is determined that the topmost element type pair is classified as matching;

confirm non-compliance of the source document if it is determined that the topmost element pair is classified as disjointed; and further:

if it is determined that the topmost element type pair is classified as intersecting, perform steps of:

identify a current child element pair of the topmost element type pair from the subtree, wherein the current child element pair comprises an element type pair from the source schema and an element type pair from the target schema;

classify the current child element pair into one of a group consisting of: matching, disjointed, and intersecting;

confirm non-compliance of the source document if the current child element pair is classified as disjointed;

process a next child element pair from the subtree if the current child element pair is classified as matching, wherein the processing step further comprises comparing element pairs; and

identify the next child element pair from the subtree as the topmost element pair if the next child element pair does not belong to a same parent node as the current child element pair.